ISSN 1561-5359. Штучний інтелект, 2016, № 2

УДК 681.03

А.Н. Химич, А.Ю., Баранов, А.В. Попов, Т.В. Чистякова
Институт кибернетики им. В.М. Глушкова НАН Украины
пр. Академика Глушкова, 40, г. Киев, 03680

ТЮНИНГОВАЯ ПРОГРАММНАЯ СИСТЕМА ДЛЯ РЕШЕНИЯ ЛИНЕЙНЫХ СИСТЕМ НА ГИБРИДНЫХ КОПЬЮТЕРАХ

A.N. Khimich, A.Ju. Baranov, A.V. Popov, T.V. Chistjakova,
V.M. Glushkov Institute of Cybernetics, NAS of Ukraine
Akademik Glushkov pr, 40, Kyiv, 03680

TUNING SOFTWARE FOR SOLVING LINEAR SYSTEMS ON HIBRID COMPUTER

Предлагается тюнинговая программная система для автоматического исследования и решения систем линейных алгебраических уравнений с приближенно заданными входными данными на гибридных компьютерных системах, обеспечивающая адаптивную настройку алгоритма, программы и конфигурации гибридной системы на свойства компьютерной модели задачи, для ее эффективного решения, с оценкой достоверности результатов. При этом, пользователи освобождаются от проблем, связанных с программированием на сложных гибридных системах.
Ключевые слова: тюнинговая программая система, математическое моделирование, приближенные данные, оценки достоверности решения, автоматическое исследование и решение задач.
An tuning software system is proposed for the automatic investigating and solving of linear algebraic systems with approximately given input data on hybrid computer, also able to adaptively adjust algorithm, program and configuration of the computer’s the hybrid system basing on the properties of a computer model of the problem for its effective solutions with the assessment of the reliability of the results. However, users are freed from the problems associated with complex programming on hybrid systems.
Keywords: tuning software system, mathematical modelling, approximate data, reliability estimates of solution, intelligent software, computer-aided investigating and solving of problems.

Введение

Математическое моделирование и связанный с ним численный эксперимент являются одними из основных средств изучения различных явлений природы и процессов во многих предметных областях: аэродинамика, экономика, экология, астрофизика, медицина и т. д. Численные эксперименты позволяют получать новые знания о тех явлениях и процессах, исследования которых с помощью натурных экспериментов наталкиваются на серьезные трудности. Однако, проведение математического моделирования на компьютере требует значительного времени и предварительных исследований (постановка прикладной задачи, создание физических, математических и дискретных моделей, разработка алгоритмов, вычислительных схем, программ решения и т.д.) [1]. Заметим, что эти исследования относятся к математическим задачам, и они осуществляются традиционными математическими средствами. Но в компьютере, после ввода данных, возникает компьютерная модель задачи, свойства которой в ряде случаев могут отличаться от свойств математической задачи [2, 3]. В некоторых случаях при решении практических задач компьютеры выдают решение, не имеющееся физического смысла. Это выясняется в ходе сопоставления данных численного и натурных экспериментов. Такая ситуация может происходить из-за ряда причин: из-за погрешностей в задании исходных данных, которые имеют место при решении прикладных задач; из-за погрешностей округлений при вычислениях; из-за отличий аксиоматики машинной математики и аксиоматики математики. Получение компьютерных решений с гарантией их достоверности является важнейшим требованием к современному программному обеспечению.

В настоящее время наблюдается определенная тенденция в развитии вычислительных систем. С одной стороны, как и прежде, продолжается рост производительности компьютеров за счет увеличения числа процессорных ядер, а с другой стороны, становятся все более популярными многоядерные компьютеры (CPU) с использованием графических процессоров (GPU) − компьютеры гибридной архитектуры, высокая производительность которых обусловлена применением вычислительных ресурсов принципиально новой архитектуры. Первое направление развития требует от алгоритмов большой степени параллелизма на однотипных процессорных ядрах, которые на программном уровне реализуются с помощью специальных систем параллельного программирования, таких, например, как MPI, OpenMP. Второе направление требует от алгоритма более сложной многоуровневой параллельной модели, которая учитывает различные архитектуры используемых вычислительных ресурсов. Кроме того, требуются навыки параллельного программирования на GPU с использованием систем OpenCL, CUDA и т. д.

Известно, что многие программные приложения из различных предметных областей зависят от успешного решения систем линейных алгебраических уравнений (СЛАУ) больших размеров, к которым сводятся решаемые задачи. Причем, на решение линейных систем тратится большая часть времени из общего времени решения задачи. Поэтому проблеме создания эффективного программного обеспечения для решения задач линейной алгебры уделяется пристальное внимание.

Существует большое количество высокопроизводительных программных средств, для решения СЛАУ на компьютерах различных параллельных архитектур, например, известные математические библиотеки стандартных программ – Intel MKL, LAPACK, MATLAB, ScaLAPACK, MAGMA и другие. Однако в этих библиотеках исследовательская функция компьютерной модели задачи и анализа результатов решения реализована частично, а учет приближенного характера модели возлагается на пользователя. При их использовании, для решения линейных систем с приближенными данными, могут возникать проблемы достоверности компьютерного решения, во избежание которых в компьютере необходимо: исследовать корректность постановки математической задачи; определить обусловленность задачи; оценить погрешность решения и т. д. Другая проблема связана с эффективным использованием вычислительных ресурсов гибридного компьютера.

Предлагается тюнинговая программная система для автоматического исследования и решения СЛАУ с приближенными данными на гибридных компьютерах, обеспечивающая адаптивную настройку алгоритма, программы и конфигурации компьютера на свойства компьютерной модели задачи для ее эффективного решения с оценкой достоверности результатов.

Методология автоматического исследования и решения линейных систем с приближенными данными на гибридном компьютере

Основными принципами рассматриваемой методологии являются: планирование вычислений с эффективным использованием вычислительных устройств; исследование свойств компьютерной модели задачи и автоматическое построение необходимого алгоритма, синтез соответствующей программы решения; анализ достоверности решения системы; обеспечение работы пользователей на компьютере со сложной архитектурой, как на однопроцессорном компьютере [4].

Созданы гибридные блочные алгоритмы прямых методов LL^T-разложения, LDL^T-разложения и LU-разложения для решения СЛАУ с плотными, ленточными, разреженными невырожденными матрицами, а также метод сингулярного разложения

(*SVD*-разложение) для нахождения нормального обобщенного решения систем с матрицами неполного ранга. На основе этих алгоритмов, реализуется компьютерное исследование математических свойств задачи с приближенными данными, а именно: существование и единственность классического решения СЛАУ, обусловленность матрицы, вырожденность матрицы, в том числе в пределах точности вычислений на компьютере. В случае симметричной матрицы исследуется ее положительная определенность. По результатам исследования автоматически выбирается необходимый алгоритм и синтезируется программа решения задачи с оценками достоверности компьютерных результатов.

Для эффективного использования ресурсов гибридного компьютера выполняются следующие действия: анализируются возможные вычислительные схемы с целью разделения исходной задачи на части (подзадачи), которые могут быть реализованы в значительной степени независимо друг от друга; устанавливается приоритетность выполнения подзадач, информационные зависимости между ними, объем каждой подзадачи; определяются необходимые или доступные для решения вычислительные устройства (процессоры и ядра CPU, мультипроцессоры GPU). Двухуровневая организация памяти гибридного компьютера предусматривает двухуровневую MPI+CUDA параллельную реализацию алгоритмов и программ. На верхнем уровне распараллеливание осуществляется между распределенной памятью вычислительных узлов, используя межпроцессорные обмены с помощью MPI. На нижнем уровне распараллеливание на GPU происходит с помощью CUDA с использованием сложной иерархичной памяти графических процессоров [5].

При реализации подзадач на CPU очень остро стоят вопросы определения необходимой топологии и оптимального количества процессов, поскольку на взаимосвязи между процессорами тратится значительная часть времени [3]. Для каждого алгоритма предусмотрено автоматическое создание виртуальной топологии межъядерных связей CPU (кольцо, решетка, тор, гиперкуб и т. д.) с помощью соответствующих функций системы MPI. Процесс − это часть программы (подзадачи), которая выполняется на одном ядре CPU, использует для своей работы часть локальной памяти, а также содержит ряд операций приема / передачи данных для организации информационного взаимодействия между другими процессами. Для каждого алгоритма автоматически выполняется распределение матриц и векторов между процессами, обеспечивая балансировку загруженности ядер и процессоров CPU. Кроме того, с целью улучшения быстродействия вычислений, процессоры CPU активно используют кэш-память.

Графический процессор устроен принципиально иначе. Большое количество потоков команд выполняется одновременно, обеспечивая массовый параллелизм вычислений и их высокое быстродействие на GPU. Однако, использовать вычислительные возможности GPU в полной мере можно только в том случае, когда задача распараллеливается на сотни исполняющих потоков, то есть, когда одна и та же последовательность математических операций применяется к большому объему данных (например, матрично-матрично/матрично-векторные операции). GPU отличается от CPU также по принципам доступа к памяти. Технология CUDA создает на GPU 6 видов памяти, каждая из которых имеет свое назначение. На рис. 1 представлена блок-схема исследования и решения СЛАУ с плотными матрицами.

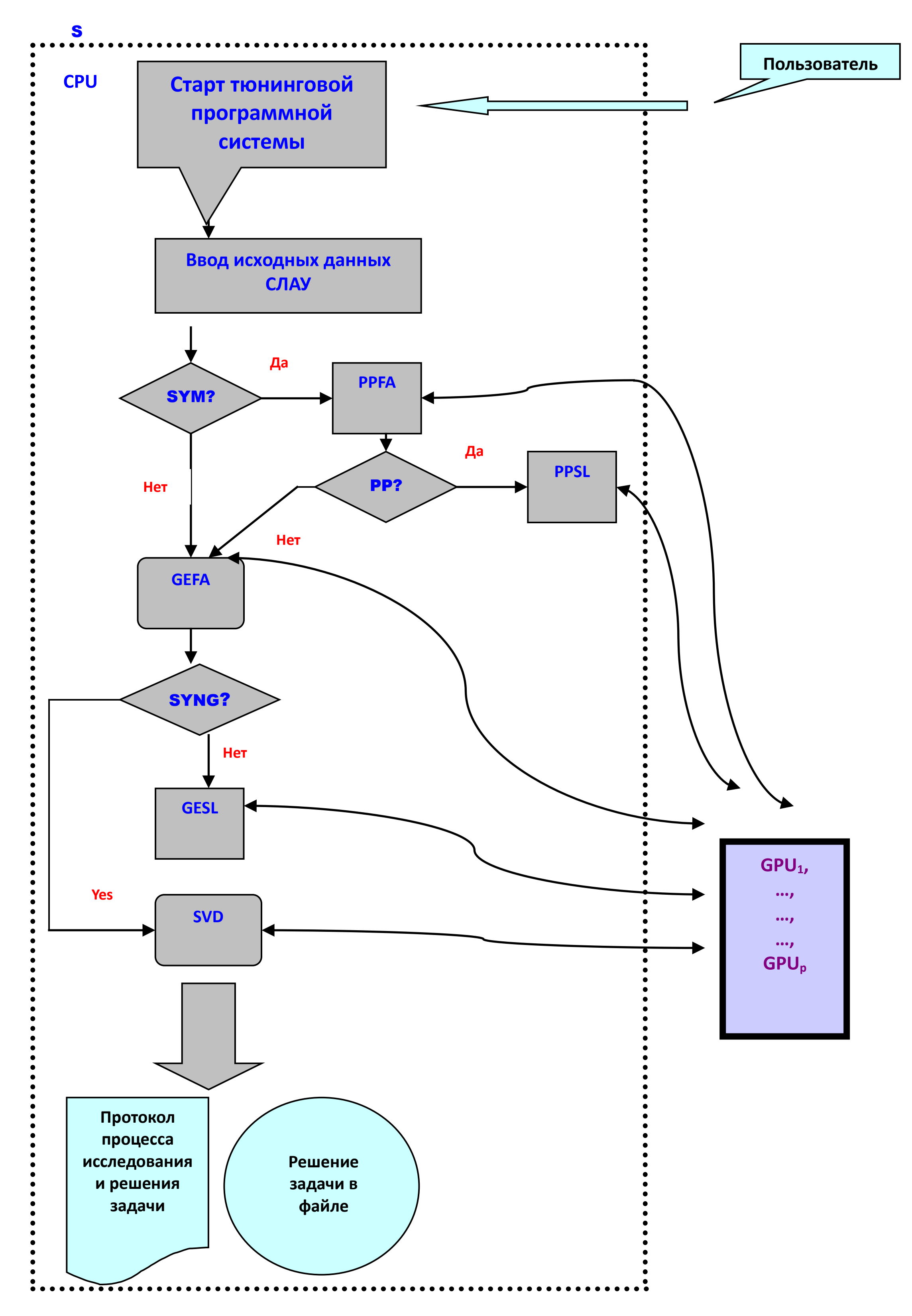


Рис 1. Фрагмент блок-схемы автоматического исследования и решения СЛАУ

Автоматическая тюнинговая программная система Inpartool_g

Автоматическая тюнинговая программная система Inpartool_g для решения СЛАУ с приближенными данными разработана и функционирует в Институте кибернетики им. В.М. Глушкова НАН Украины на суперкомпьютерах гибридной архитектуры Инпарком_g и СКИТ-4 [4, 6]. Архитектура Inpartool_g схематичеки представлена на рис. 2.

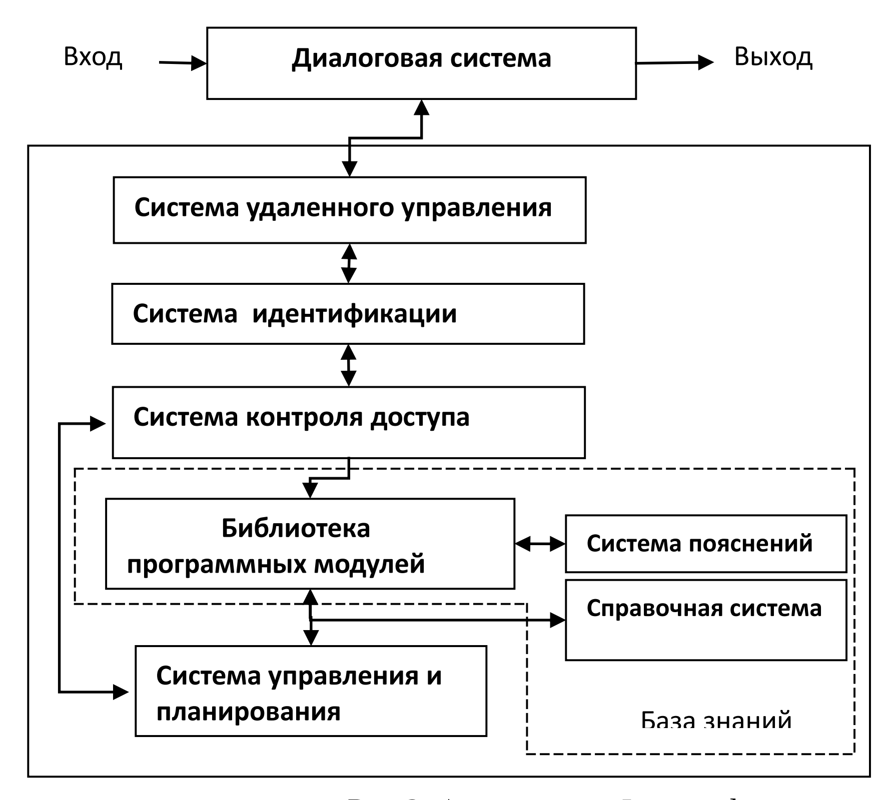


Рис 2. Архитектура Inpartool_g

Программная реализация Inpartool_g выполнена с применением клиент-серверной архитектуры. Доступ к Inpartool_g осуществляется по паролю, имени пользователя и электронного сертификата как локально, так и в Internet.

Программные модули в Inpartool_g используются для автоматического построения необходимого алгоритма и соответствующей топологии компьютера, а также для автоматического синтеза необходимой программы решения задачи на основе выявленных свойств компьютерной модели задачи. Связь между модулями устанавливается по данным и по управлению.

Принципы автоматического исследования и решения линейных систем на компьютере с автоматическим анализом достоверности результатов определяют следующие требования к системе управления и планирования в Inpartool_g:

1) анализ исходных данных, полученных от пользователя, и преобразование их в первичные знания о свойствах компьютерной модели задачи;
2) возможность хранить, обрабатывать и применять полученные знания о свойствах задачи при планировании вычислений;
3) автоматическое построение гибридных алгоритмов и синтез программ исследования и решения задач;
4) установление необходимого количества CPU и GPU, построение компьютерной топологии для эффективного решение задачи;

5) распределение и сбор данных о задаче между процессорами CPU и GPU.
6) выдача и сохранение результатов исследования и решения для последующего их использования, объяснения и визуализации.

Система объяснений отвечает на вопросы: как было получено решение задачи, почему был выбран такой путь процесса исследования свойств; выдает полученное решение с оценками достоверности или объясняет причины отказа в его получении.

Справочная система позволяет пользователю получить необходимую информацию, чтобы решить его задачу с помощью Inpartool_g: о функциональных возможностях, порядке работы, вводе исходных данных, глоссарий используемых терминов из предметной области и т. д.

С помощью диалоговых средств в Inpartool_g осуществляется взаимодействие с пользователем, а именно: постановка задачи и ввод исходных данных; запуск задачи на выполнение; взаимодействие (по желанию) в ходе вычислительного процесса; визуализация полученных результатов; доступ к блоку объяснений; получение информационно-справочных сведений и помощи на каждом этапе работы.

Диалоговая система дает возможность пользователям осуществлять постановку исходной задачи, а последовательность действий для получения ее решения автоматически определяется самим программным средством. Порядок взаимодействия Inpartool_g и пользователя устанавливается посредством главного меню. Его структура и основные пункты естественны и привычны для пользователя, поскольку присущи многим диалоговым системам. При вводе исходных данных пользователь заполняет оконные формы при помощи подсказок и инструкций или отвечает на вопросы Inpartool_g. На рис. 3 представлено окно ввода исходных данных линейной системы в Inpartool_g.

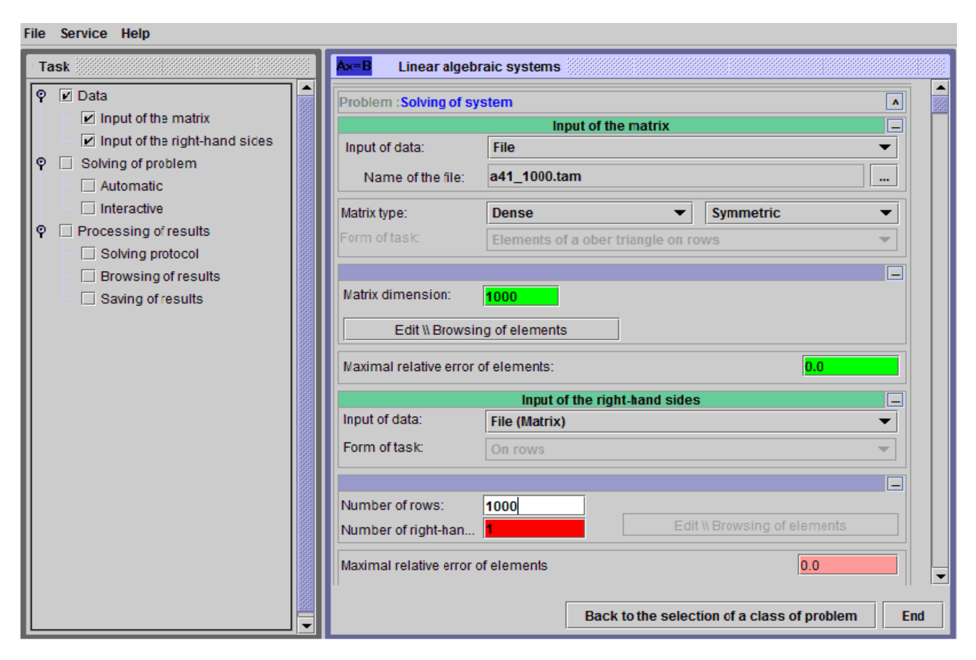


Рис 3. Окно ввода исходных данных в Inpartool_g

Решение задачи может реализовываться автоматически, когда ее исследование и решение выполняются без вмешательства пользователя, и интерактивно, когда на отдельных или на всех этапах исследования и решения задачи возможно участие пользователя (рис. 4).

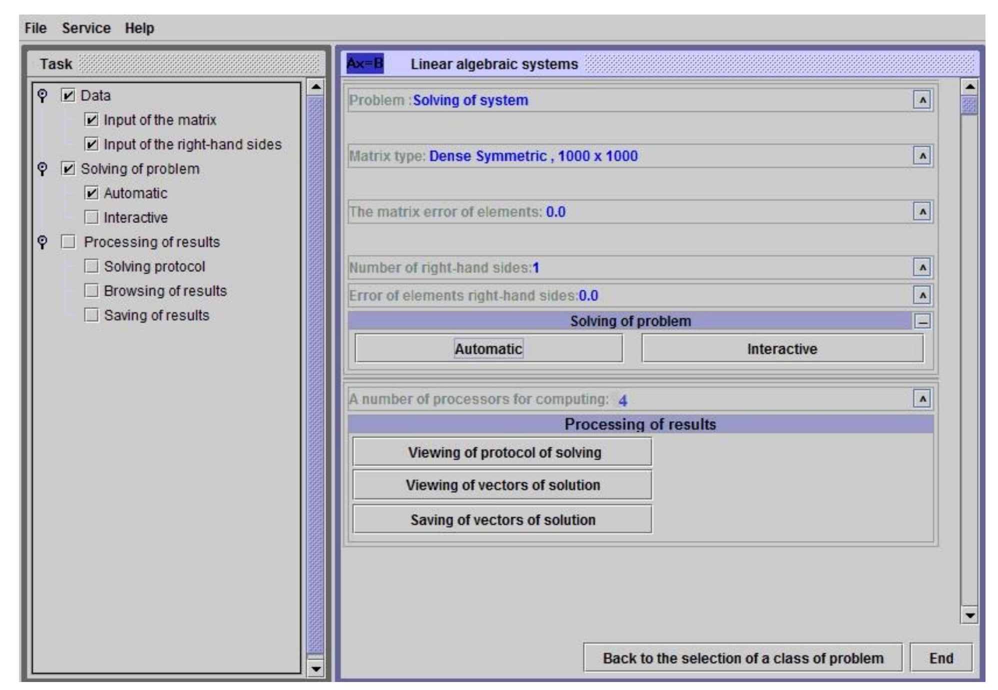


Рис 4. Окно выбора режима решения задачи в Inpartool_g

По окончании решения задачи в Inpartool_g пользователь получает в протоколе решения сообщения о ходе решения, выявленных свойствах задачи, достоверности полученных результатов решения или причинах отказов в решении.

Пример исследования и решения СЛАУ в Inrartool_g

Рассмотрим на конкретном примере как работает Inrartool_g на компьютере гибридной архитектуры СКІТ-4.

Задача. Исследовать и решить в Inpartool_g систему $Ax = b$, где элементы симметричной матрицы вычисляются по формулам: $A = (a_{ij})$, $i, j = 1$ □ n, $n = 3w + 1$, $w = 1, 2, \ldots$; $a_{ii} = n - i$, $a_{ij} = n + 1 - \max(i, j)$.

Матрица A имеет вид:

$$A = \begin{pmatrix} n-1 & n-1 & n-2 & \ldots & 2 & 1 \\ n-1 & n-2 & n-2 & \ldots & 2 & 1 \\ n-2 & n-2 & n-3 & \ldots & 2 & 1 \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ 2 & 2 & 2 & \ldots & 1 & 1 \\ 1 & 1 & 1 & \ldots & 1 & 0 \end{pmatrix}.$$

Элементы правой части вычисляются по формулам:

$b = \{b\}_1^n$, $b_i = n - i$, если i □ 2; $b_i = n + 1 - i$, если $i > 2$.

Точное решение системы: $x = (0 \quad 1 \quad 0 \quad \ldots \quad 0)^T$.

Ниже приведен листинг протокола автоматического исследования и решения СЛАУ в Inpartool_g.

```
PROBLEM:
     solving of system of linear algebraic equations with symmetic positive defined matrix

  Input Parameters:
    - the number of rows                  = 1000
    - the number of columns               = 1000
    - the number of right hand sides      = 1
    - maximal relative errors
        of matrix elements                = 0.000e+00
        of right hand sides elements      = 0.000e+00

Process of investigating and solving

  Method:
     Choletsky decomposition

  RESULTS:
     !!! THE MATRIX IS NOT POSITIVE DEFINED

  Method:
    Gauss elimination with partial pivoting

  RESULTS:
     !!! THE MATRIX IS MACHINE-SINGULAR

  Method:
    singular value decomposition of a general matrix

     RESULTS:  SOLUTION WAS CALCULATED

  the first 12 components of solution (vector 1) are:

  4.3032264e-10   1.0000000e+00   1.0771629e-09  -3.0017623e-10
 -8.4157109e-11  -3.6427290e-11  -2.1761434e-10   7.6932562e-11
  6.5265384e-11   1.5581740e-10  -3.4085320e-11  -1.9264943e-10

The vecror(s) of solution are successfully stored in the file result.out

  Error estimations:   4.99145e-08

  Properties:
  - estimation of conditional number: 7.49316e+07
  - matrix rank: 999 (999)

  Number of processors: 1
```

Как видно из протокола, пользователь указал, что исходные данные системы заданы точно. Однако в Inpartool_g проводится исследование системы, введенной в компьютер, с учетом ошибок округлений при вводе. Поскольку матрица является симметричной, то пробным алгоритмом для исследования задачи был выбран гибридный алгоритм (удвоенная разрядность) LL^T-факторизации на одном CPU с использованием одного GPU, как наиболее экономичный по использованию вычислительных ресурсов и времени решения для таких матриц. Однако, в процессе

исследования СЛАУ, этим алгоритмом матрица оказалась не положительно-определенной и Inpartool_g автоматически продолжил исследование задачи гибридным алгоритмом *LU*-факторизации на одном CPU с использованием одного GPU. В процессе исследования системы этим алгоритмом матрица оказалась машинно-вырожденной. Автоматически была синтезирована программа нахождения обобщенного решения гибридным *SVD*-алгоритмом с использованием одного CPU и одного GPU-ускорителя. Задача была решена (в протоколе мы имеем 12 первых компонентов решения). Вычислены оценка достоверности полученного решения, а также оценка спектрального числа обусловленности, ранг матрицы. Весь вектор решения в бинарном виде автоматически сохранен в файле result.out.

Проведем анализ прохождения процесса решения этой задачи с помощью высоко-производительных программ из библиотек Lapack, ScaLAPACK, Magma. Программы этих библиотек ориентированы на точно заданные исходные данные. Чтобы правильно определить необходимую программу решения этой задачи, пользователю необходимо знать свойства матрицы системы (положительно определенная, невырожденная или вырожденная), которая введена в компьютер. Заранее это сделать зачастую невозможно из-за приближенного характера исходных данных и ошибок округлений при их вводе в компьютер. Таким образом, пользователю потребуется потратить дополнительное время на самостоятельное исследование задачи. Для этого ему придется осуществить несколько постановок задач, используя различные программы. Анализ достоверности полученного решения также возлагается на пользователя.

Выводы

Многие научно-технические задачи сводятся к решению СЛАУ больших размеров, для решения которых требуется эффективное использование ресурсов гибридных компьютеров. В условиях приближенных данных свойства компьютерных моделей априори не известны. Поэтому необходимы такие подходы к созданию программного обеспечения для гибридных компьютеров, которые бы обеспечивали пользователей достоверным компьютерным решением при эффективном использовании CPU и GPU.

Предлагается автоматическая тюнинговая программная система Inpartool_g для исследования и решения СЛАУ на гибридных компьютерах с функцией адаптивной настройки алгоритма, программы и архитектуры гибридной системы на свойства решаемой задачи для ее эффективного решения с оценкой достоверности компьютерных результатов. Использование Inpartool_g позволяет пользователю: работать на компьютере гибридной архитектуры так же, как и на однопроцессорном компьютере; сократить время постановки и решения задачи примерно на два порядка; гарантировать достоверность полученных машинных результатов.

Литература

1. Сергиенко И.В. Интеллектуальные технологии высокопроизводительных вичислений / И.В Сергиенко., И.Н Молчанов., А.Н. Химич. Кибернетика и системный анализ, 2010. – № 5. –С. 64 – 76.
2. Молчанов И.Н. Машинные методы решения прикладных задач. Алгебра, приближение функций. / И.Н. Молчанов. – Київ: Наукова думка, 1987. – 288 с.
3. Химич А.Н. Параллельные алгоритмы решения задач вычислительной математики. / А.Н Химич, И.Н. Молчанов, А.В. Попов, Т.В. Чистякова, М.Ф. Яковлев. – Киев: Наукова думка, – 2008. – 247 с.
4. Химич А.Н. Численное программное обеспечение MIMD–компьютера Инпарком. / А.Н. Химич. И.Н Молчанов., В.И. Мова и др. – Киев: Наукова думка, – 2007. – 222 с.
5. Боресков А.В. Основы работы с технологией CUDA. / А.В. Боресков, А.А. Харламов. - М.: Пресс, 2010. - с. 232.
6. Суперкомпьютер ИК НАН Украины СКИТ. // [Электр. Ресурс]. – Режим доступа: http://icybcluster.org.ua.

Literatura

1. Sergienko I.V. Intellektual'nye tekhnologii vysokoproizvoditel'nykh vychislenii / I.V. Sergienko, I.N. Molchanov, A.N. Khimich. Kibernetika i sistemnyi analiz, 2010, N 5: – S. 64 – 76.
2. Molchanov I.N. Machinye metody reshenija prikladnyh sadach. Algebra, priblizhenie funkcij. / I.N. Molchanov. − Kyiv: Naukova dumka, 1987. – 288 s.
3. Khimich A.N. Parallel'nye algoritmy reshenija zadach vychislitel'noi matematiki. / A.N. Khimich, I.N Molchanov, A.V Popov, T.V Chistjakova, M.F. Jakovlev. − Kyiv: Naukova dumka, 2008. – 247 s.
4. Khimich A.N. Chislennoe programmnoe obespechenie intellektual'nogo MIMD-komp'jutera Inparcom / A.N. Khimich, I.N Molchanov, V.I. Mova et al. − Kyiv: Naukova dumka, 2007. – 222 s.
5. Boreskov A.V. Basics with CUDA technology. / A.V. Boreskov, A.A. Kharlamov. − M.: Press, 2010. − 232 s. Supercomputer of IC NAN Ukraine SCIT. // [Elektr. Resurs]. – Rezhym dostupu: // http://icybcluster.org.ua.

RESUME

A.N. Khimich, A.Ju. Baranov, A.V. Popov, T.V. Chistjakova

Tuning software for solving linear systems on hybrid computer

Many scientific and technical problems are reduced to solving systems of linear algebraic equations of large size for efficient solutions that require the use of powerful computing resources. Currently, they are becoming more popular hybrid computers - multi-core processors with GPUs, performance which is caused by the use of the computing resources of a fundamentally new architectures.

In mathematical modeling of tasks on the computer after a data entry problem arises computer model, the properties of which in some cases may be different from those of a mathematical problem. In some cases, in solving problems of engineering and science computer gives a decision, which has no physical meaning. This situation can occur due of several reasons: due of errors in setting the initial data that take place in the solution of applications; because of rounding errors in calculations; because to differences between the axioms of mathematics and machine axiomatic mathematics.

There are a large number of high-performance application software for solving systems of linear algebraic equations on computers of different architectures. However, the research function of a computer model of the problem and solutions results in these libraries is implemented partially, keeping the approximate nature of the model with the user. To effectively use this software, you must know the properties of the problem computer model, mathematical and technical features of the computer on which the problem is solved, master the skills of parallel programming.

Inpartool_g tuning software system proposed for the automatic research and solving linear systems with different data structure that implements: planning calculations on necessary computational resources of a hybrid computer, study and solution of the problem with approximate data, the analysis of the reliability of the solutions. The user escapes from the problems of parallelizing tasks on a hybrid computer.

Надішла до редакції 02.09.2016

